IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPEAL BRIEF

Applicant:	Eickemeyer, *et al.*	Docket No.:	ROC920020128US1
Serial No.:	10/717,747	Group Art Unit:	2183
Filed:	11/20/03	Examiner:	Johnson, Brian P

TITLE: MULTITHREADED PROCESSOR AND METHOD FOR SWITCHING THREADS

Mail Stop APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir/Madam:

This appeal is taken from the Examiner's final rejection, set forth in the Office Action dated 2/19/2008 of Appellant's claims 1-22. Appellant's Notice of Appeal under 37 C.F.R. § 1.191 was filed via EFS-Web on 05/19/2008.

REAL PARTY IN INTEREST

International Business Machines Corporation is the Real Party in Interest.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences for this patent application.

STATUS OF CLAIMS

Claims 1-22 were originally filed in this patent application. In response to the office action dated 9/14/2006, an RCE and amendment was filed on 09/27/2006 to amend claims 1, 3, 5, 7, 8, 9 and 14. In response to the office action dated 12/13/2006, an amendment was filed 3/13/2007 to amend claim 8. In an office action dated 09/05/2007, claims 1-2 and 9-13 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,348,671 to Doing *et al.* (hereinafter "Doing") in view of U.S. Patent No. 5,933,627 to Parady and claims 3-6, 8, 14-19, 21 and 22 were rejected under 35 U.S.C. §103(a) as being unpatentable over Doing in view of Parady and further in view of U.S. Patent Application Publication No. 2003/0135711 to Shoemaker *et al.* (hereinafter "Shoemaker"). Claims 7 and 20 were rejected under 35 U.S.C. §103(a) as being unpatentable over Doing in view of Parady, Shoemaker and U.S. Patent No. 6,314,511 to Levy *et al.* (hereinafter "Levy"). Claims 7 and 20 were also rejected under 35 U.S.C. §103(a) as being unpatentable over Doing in view Parady and Shoemaker. In response to this last rejection, Appellant filed a Request for Reconsideration on 11/30/2007. In response, the examiner issued a final office on 02/19/2008 reiterating the previous rejections. No claim was allowed. Claims 1-22 are currently pending and are at issue in this appeal.

STATUS OF AMENDMENTS

The amendment filed 09/27/2006 has been entered. The amendment filed 3/13/2007 was non-compliant and was not entered. In response to a notice of non-compliant amendment on 5/24/2007, the amendment filed on 06/04/2007 was entered. Therefore, the claims at issue in this appeal are claims 1-22 as amended in the amendment dated 06/04/2007.

SUMMARY OF CLAIMED SUBJECT MATTER

Claim 1 recites an integrated circuit processor comprising a first instruction buffer corresponding to a primary thread (Figure 2, 212; page 5, line 5); a second instruction buffer corresponding to a backup thread (Figure 2, 214; page 5, lines 5-6); a thread switch mechanism that detects when the primary thread stalls, and in response thereto, swaps instructions stored in the first instruction buffer with instructions stored in the second instruction buffer (Figure 3, 320, page 6, lines 5-6).

Claim 5 recites an integrated circuit processor comprising: a first primary instruction buffer corresponding to a first primary thread (Figure 2, 212; page 5, line 5); a second primary instruction buffer corresponding to a second primary thread (Figure 2, 222; page 6, lines 5-6); wherein the first and second primary threads simultaneously issue instructions for execution (p. 5 line 16); a first backup instruction buffer (Figure 2, 214; page 5, lines 5-6); a second backup instruction buffer (Figure 2, 224; page 5, line 7); a thread switch mechanism that detects when one of the first and second threads stalls, and in response thereto, swaps instructions stored in one of the first and second primary instruction buffers corresponding to the stalled thread with instructions stored in one of the first and second backup instruction buffers (Figure 3, 320, page 6, lines 4-9).

Claim 8 recites an integrated circuit processor comprising a first primary instruction buffer corresponding to a first primary thread (Figure 2, 212; page 5, line 5); a second primary instruction buffer corresponding to a second primary thread (Figure 2, 222; page 6, line 5-6); wherein the first and second primary threads simultaneously issue instructions for execution (p. 5 line 16); a first backup instruction buffer (Figure 2, 214; page 5, lines 5-6); a second backup instruction buffer (Figure 2, 224; page 5, line 7); a thread switch mechanism that detects when the first thread stalls, and in response thereto, swaps instructions stored in the first primary instruction buffer with instructions stored in the first backup instruction buffer, and begins issuing from the first primary instruction buffer, and that detects when the second thread stalls, and in response thereto, swaps

instructions stored in the second primary instruction buffer with instructions stored in the second backup instruction buffer, and begins issuing from the second primary instruction buffer (Figure 5; page 6, lines 4-12).

Claim 9 recites a method for switching between a first thread of execution and a second thread of execution in a multithreaded processor, the method comprising the steps of: (A) providing a first instruction buffer corresponding to the first thread (Figure 2, 212; page 5, line 5); (B) providing a second instruction buffer corresponding to the second thread (Figure 2, 214; page 5, lines 5-6); (C) swapping instructions stored in the first instruction buffer with instructions stored in the second instruction buffer (Figure 3; page 6, lines 4-9).

Claim 18 recites a method for switching between first and second threads of execution in a multithreaded processor, the method comprising the steps of (A) providing a first primary instruction buffer corresponding to the first thread (Figure 2, 212; page 5, line 5); (B) providing a second primary instruction buffer corresponding to the second thread (Figure 2, 222; page 5, lines 5-6); (C) providing a first backup instruction buffer corresponding to a first backup thread (Figure 2, 214; page 5, lines 5-6); (D) providing a second backup instruction buffer corresponding to a second backup thread (Figure 2, 224; page 5, line 7); (E) simultaneously issuing instructions from the first primary instruction buffer and from the second primary instruction buffer (p. 5 line 16); and (F) detecting when one of the first and second primary threads stalls, and in response thereto, swapping instructions stored in one of the first and second primary instruction buffers corresponding to the stalled thread with instructions stored in one of the first and second backup instruction buffers (Figure 5; page 6, lines 4-12).

Claim 21 recites a method for switching between threads of execution in a multithreaded processor, the method comprising the steps of (A) providing a first primary instruction buffer corresponding to the first thread (Figure 2, 212; page 5, line 5); (B)

providing a second primary instruction buffer corresponding to the second thread (Figure 2, 222; page 5, lines 5-6); (C) providing a first backup instruction buffer corresponding to a first backup thread (Figure 2, 214; page 5, lines 5-6); (D) providing a second backup instruction buffer corresponding to a second backup thread (Figure 2, 224; page 5, lines 7); (E) simultaneously issuing instructions from the first primary instruction buffer and from the second primary instruction buffer (p. 5 line 16); and (F) detecting when the first threads stalls, and in response thereto, swapping instructions stored in the first primary instruction buffer with instructions stored in the first backup instruction buffer, and issuing instructions from the first primary instruction buffer (Figure 5; page 6, lines 4-9).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The following grounds of rejection are presented for review on this Appeal:

1. **Whether claims 1-2 and 9-13 are unpatentable under 35 U.S.C. §103(a) over Doing in view of Parady.**

2. **Whether claims 3-6, 8, 14-19, 21 and 22 are unpatentable under 35 U.S.C. §103(a) over Doing in view of Parady and further in view of Shoemaker.**

3. **Whether claims 7 and 20 are unpatentable under 35 U.S.C. §103(a) over Doing in view of Parady and Shoemaker and further in view of Levy.**

4. **Whether claims 7 and 20 are unpatentable under 35 U.S.C. §103(a) over Doing in view of Parady and Shoemaker and further in view of Parady.**

ARGUMENT

Issue 1: Whether claims 1-2 and 9-13 are unpatentable under 35 U.S.C. §103(a) over Doing in view of Parady.

Claims 1 and 2

For the claim limitation "swaps instructions stored in the first instruction buffer with instructions stored in the second instruction buffer", the Examiner cites Parady (col. 5, lines 38-43 and Fig. 7). In the alternative embodiment shown in Figures 6 and 7, there are shadow registers in the register file that is associated with the integer execution unit. With reference to the register file 184 and the shadow register files 186, Parady teaches the "data switch 192 can be used to route data to and from the shadow registers" (col. 5, lines 40-41, emphasis added). Parady further teaches in the preceding paragraph that "it may be more economical to swap the thread data in and out of the four, single or dual-ported shadow register files" (col. 5, lines 32-34). Thus the sections of Parady that the Examiner has relied on for teaching to swap instructions actually teaches to swap data.

In the office action dated 9/5/2007, the Examiner argues that it is obvious to incorporate the swapping of state data from the shadow registers as taught in Parady with the instruction buffers in Doing to exchange "not just various context information but instructions themselves during a thread switch" (page 4, lines 4-5). Support for this conclusion draws on the recent Supreme Court case of KSR v. Teleflex, 550 U.S. ____ (2007), where the Supreme Court stated:

> "When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense."

In applying the Supreme Court's teaching above, the Examiner noted that there are primarily two options available to one of ordinary skill in the art. The second option noted by the Examiner is the combination of Parady and Doing set forth by the Examiner. This is simply not how the Supreme Court applied this rationale in KSR. In the Supreme Courts words, there was "good reason to pursue **known options**", (emphasis added). In KSR v. Teleflex, the patent that was found to be obvious combined the known option of a sensor on a gas pedal, with the known option of an adjustable gas petal with a fixed pivot. So in KSR, the combined elements of the prior art were both known to be options of a gas pedal. In this case, the second option the Examiner identifies was not known. The second option is a combination of the prior art that the Examiner only concludes would have been known. In fact, the Examiner's reasoning looks more like hindsight, where the Examiner uses the Appellant's application as a blueprint to supply the "known option" for the KSR analysis from a combination of the prior art. The Examiner has applied the KSR analysis beyond the manner described by the Supreme Court.

The Supreme Court in KSR further cited Sakraida v. AG Pro, Inc., 425 U. S. 273 (1976), where the Court derived from the precedents the conclusion that when a patent simply arranges old elements with each performing the same function it had been known to perform, and yields no more than one would expect from such an arrangement, the combination is obvious. Id., at 282. In the present case, when the shadow registers of Parady are combined with Doing in the Examiner's combination, they do not perform the same function, in the same way, to produce a similar result as taught in the prior art. The swapping of data in the shadow registers taught by Parady does not perform the same function as the corresponding structure in the claimed invention and does not yield the same results. Swapping data in shadow registers is not the same function as performed by the thread switch mechanism. The thread switch mechanism detects when the primary thread stalls, and in response thereto, swaps instructions stored in the first instruction buffer with instructions stored in the second instruction buffer. The shadow registers do

not swap instructions so they are operating in a different way to produce a different result. Parady does not teach or suggest the invention as argued by the Examiner. The combined art does not perform the same function in the same way to provide a similar result. The Examiner has failed to establish a prima facie case.

In the Examiner's response dated 2/19/08, the Examiner argues against the Appellant as follows:

> "Examiner notes that instructions are data – just a particular kind of data. They contain are [sic] elements within a processor containing 1s and 0s that indicated how the processing system should operate. Applicant believes that, while resister data swapping is known in the art, instruction data swapping ventures into the unknown. There is no support for this claim. The two are analogous and equally obvious."

Appellant believes the Examiner has made conclusions that are not supported by the cited art. Yes, appellant believes the instruction swapping ventures into the unknown. And there is support for this statement since the Examiner has not shown it in the prior art but relies on Appellant's application to piece together the cited art. While instructions may be a type of data when stored in the computer, it does not mean that it would be obvious to modify the register data swapping hardware in the prior art to be able to handle instructions. Data and instructions may be made of the same elements but they are not the same. A hammer and a hand saw are typically made of the same elements, but they are used differently and not interchangeable. Similarly, data and instructions are not the same and not interchangeable as the Examiner's logic suggests. The Examiner's statements to the contrary, the cited art must show more than remotely similar hardware used for a different purpose to get a different result. It is the Examiner's burden to establish a prima facie case. The Appellant has shown that the Examiner's conclusory statements are lacking in factual backing and logical reasoning. Appellant respectfully requests the examiner's rejection of claims under 35 U.S.C. §103(a) be reversed.

The Examiner's argument states the motivation to combine, or the "good reason" is the advantage shown in Parady of saving cost by using less silicon for read ports. This

motivation or good reason to combine does not relate to the invention herein. There are no read ports in the combination that are being reduced to realize the cost savings. The motivation to combine must motivate one of ordinary skill in the art to combine the art in the manner claimed. The Examiner has not shown a proper and relevant motivation to combine the cited art in the manner claimed. In integrated circuits, there is almost always a tradeoff between speed and silicon real estate or die size. The claimed invention trades more silicon for increased speed by increasing the number of instruction buffers. The claimed invention does not save silicon as suggested by the Examiner's argument for motivation. The Examiner has failed to establish a prima facie case.

Claim 2 depends on claim 1, which is allowable for the reasons given above. Claim 2 is therefore allowable as depending on an allowable independent claim. Appellant respectfully requests the examiner's rejection of claims under 35 U.S.C. §103(a) be reversed.

Claims 9-13

In the rejection of claim 9, the Examiner cites column 7, line 52 to column 8, line 3 of Doing for the claim limitation of swapping information stored in the first instruction buffer with information stored in the second instruction buffer. The cited section of Doing describes changing the active thread. While it is evident that Doing does swap the contents of some registers that are associated with the thread buffer, the contents of the thread buffer are not swapped. The express teachings of Doing indicate the contents of the sequential hold buffer and the thread switch buffer are not swapped. In Doing, when there is a thread switch, the inactive thread becomes the active thread, and the active thread becomes the inactive thread as taught on column 14, lines 22 through 24. When a thread is switched from inactive to active, there is no swapping or moving of instructions between the sequential hold buffer and the thread switch buffer. To the contrary, the

decode/dispatch logic 206 simply chooses a different buffer as the source of the next instruction.

In contrast to Doing, the claims clearly distinguish that the information that is swapped in the instruction buffers comprises instructions. As described above with reference to claim 1, Parady also does not teach or suggest swapping the instructions. Doing and Parady do not teach or suggest to swap the instructions in the instruction buffers as claimed herein.

Claims 10-13 depend on claim 9, which is allowable for the reasons given above. Claims 10-13 are therefore allowable as depending on an allowable independent claim. Appellant respectfully requests the examiner's rejection of claims under 35 U.S.C. §103(a) be reversed.

Issue 2: **Whether claims 3-6, 8, 14-19, 21 and 22 are unpatentable under 35 U.S.C. §103(a) over Doing in view of Parady and further in view of Shoemaker.**

Claims 3-6, 8, 14-19, 21 and 22

Claims 3-4 depend on claim 1, which is allowable for the reasons given above. The arguments for claim 1 are included here by reference. Further, claims 3-4 include additional limitations novel over the prior art. With regards to claims 3-4, the Examiner adds the Shoemaker reference for the concept of multiple threads (more than 2). In Shoemaker, the multiple threads are threads that are all available to be selected (paragraph 6). These multiple threads could be considered as stated by the Examiner, simply replicated portions of Doing. However, the claimed invention is not simply replicated portions of Doing. In theses claims, the primary threads swap instructions with

secondary threads when there is a stall. The cited art does not teach or suggest swapping instructions for threads from a secondary thread to a primary thread when a thread stalls. Appellant respectfully requests the board to reverse the examiner's rejection of claims 3-4 under 35 U.S.C. §103(a).

Independent claims 5, 8 and 18 include similar limitations as claims 3-4 and are allowable for the reasons given above. Claims 6, 14-17, 19 and 21-22 depend on claims 5, 8 and 19 respectively. Claims 3-4, 6, 14-17, 19 and 21-22 are therefore allowable as depending on allowable independent claims. Appellant respectfully requests the examiner's rejection of these claims under 35 U.S.C. §103(a) be reversed.

Issue 3: Whether claims 7 and 20 are unpatentable under 35 U.S.C. §103(a) over Doing in view of Parady and Shoemaker and further in view of Levy.

Claims 7 and 20

The Examiner rejected claims 7 and 20 under 35 U.S.C. §103 as being unpatentable over Doing in view of Parady in view of Shoemaker and further in view of Levy. Appellant asserts the cited art does not teach or suggest the claimed invention as amended herein. Claims 7 and 20 depend on claims 5 and 18 respectively, which are allowable for the reasons given above. Therefore claims 7 and 20 are allowable as depending on an allowable claim. For the claim limitation of "a pool of backup registers," the Examiner cites Levy's use of a pool of registers to be used for register renaming during a context switch. Again, the structure in the cited art does not teach or suggest to use a pool of registers in the manner claimed. While there is a pool of registers in Levy, they do not provide the same function in the same way to provide a similar result. The pool of registers in Levy is not used to hold instructions. They do not

function in a similar way to allow swapping of an instruction stream to provide continued operation when there is stalled thread.

The Examiner's motivation to combine in this rejection suffers from the same problem discussed above. The motivation is directed to a structure in the prior art and does not motivate one of ordinary skill in the art to combine the art in the manner claimed. Appellant respectfully requests the board to reverse the examiner's rejection of claims 7 and 20 under 35 U.S.C. §103(a).

Issue 4: Whether claims 7 and 20 are unpatentable under 35 U.S.C. §103(a) over Doing in view of Parady and Shoemaker and further in view of Parady.

Claims 7 and 20

The Examiner rejected claims 7 and 20 under 35 U.S.C. §103 as being unpatentable over Doing in view of Shoemaker and further in view of Parady. Claims 7 and 20 depend on claims 5 and 18 respectively, which are allowable for the reasons given above. Therefore claims 7 and 20 are allowable as depending on an allowable claim. Further, Appellant asserts the cited art does not teach or suggest the claimed invention as amended herein. In this rejection, the Examiner relies on Parady to show the pool of registers as shown in Figure 7. This rejection is confusing since these same structures (shadow registers in Figure 7) were used in the rejection of the instruction buffers in the base claim (claim 5). This rejection also suffers from the same problems as discussed above. Appellant respectfully requests the board to reverse the examiner's rejection of claims 7 and 20 under 35 U.S.C. §103(a).

CONCLUSION

Claims 1-22 are addressed in this Appeal. For the numerous reasons articulated above, appellant maintains that the rejections of claims 1-22 are erroneous.

Appellant respectfully submits that this Appeal Brief fully responds to, and successfully contravenes every ground of rejection, and respectfully requests that the final rejection be reversed and that all claims in the subject patent application be found allowable.

Respectfully submitted,

/bretjpetersen/
Bret J. Petersen
Reg. No. 37,417

MARTIN & ASSOCIATES, L.L.C.
P.O. Box 548
Carthage, MO 64836-0548
(417) 358-4700; Fax (417) 358-5757

CLAIMS APPENDIX

1. An integrated circuit processor comprising:
a first instruction buffer corresponding to a primary thread;
a second instruction buffer corresponding to a backup thread;
a thread switch mechanism that detects when the primary thread stalls, and in response thereto, swaps instructions stored in the first instruction buffer with instructions stored in the second instruction buffer.

2. The integrated circuit processor of claim 1 wherein execution of the backup thread occurs after the swap by executing at least one instruction in the first instruction buffer.

3. The integrated circuit processor of claim 1 further comprising:
a third instruction buffer corresponding to a second primary thread;
a fourth instruction buffer corresponding to a second backup thread;
wherein the thread swap mechanism further detects when the second primary thread stalls, and in response thereto, swaps instructions stored in the third instruction buffer with instructions stored in the fourth instruction buffer.

4. The integrated circuit processor of claim 3 wherein the first and second primary threads simultaneously issue instructions for execution.

5. An integrated circuit processor comprising:

a first primary instruction buffer corresponding to a first primary thread;

a second primary instruction buffer corresponding to a second primary thread;

wherein the first and second primary threads simultaneously issue instructions for execution;

a first backup instruction buffer;

a second backup instruction buffer;

a thread switch mechanism that detects when one of the first and second threads stalls, and in response thereto, swaps instructions stored in one of the first and second primary instruction buffers corresponding to the stalled thread with instructions stored in one of the first and second backup instruction buffers.

6. The integrated circuit processor of claim 5 wherein the thread switch mechanism:

(1) detects when the first primary thread stalls, and in response thereto, swaps the first primary instruction buffer with the first backup instruction buffer; and

(2) detects when the second thread stalls, and in response thereto, swaps the second primary instruction buffer with the second backup instruction buffer.

7. The integrated circuit processor of claim 5 wherein the first and second backup instruction buffers are part of a pool of backup instruction buffers, wherein instructions in any backup instruction buffer in the pool may be swapped with instructions in the first primary instruction buffer, and wherein instructions in any backup instruction buffer in the pool may be swapped with instructions in the second primary instruction buffer.

8. An integrated circuit processor comprising:

a first primary instruction buffer corresponding to a first primary thread;

a second primary instruction buffer corresponding to a second primary thread;

wherein the first and second primary threads simultaneously issue instructions for execution;

a first backup instruction buffer;

a second backup instruction buffer;

a thread switch mechanism that detects when the first thread stalls, and in response thereto, swaps instructions stored in the first primary instruction buffer with instructions stored in the first backup instruction buffer, and begins issuing from the first primary instruction buffer, and that detects when the second thread stalls, and in response thereto, swaps instructions stored in the second primary instruction buffer with instructions stored in the second backup instruction buffer, and begins issuing from the second primary instruction buffer.

9. A method for switching between a first thread of execution and a second thread of execution in a multithreaded processor, the method comprising the steps of:

(A) providing a first instruction buffer corresponding to the first thread;

(B) providing a second instruction buffer corresponding to the second thread;

(C) swapping instructions stored in the first instruction buffer with instructions stored in the second instruction buffer.

10. The method of claim 9 wherein step (C) is performed when switching between the first thread and the second thread is required.

11. The method of claim 9 wherein step (C) is performed when the first thread stalls.

12. The method of claim 9 wherein step (C) is performed when the second thread stalls.

13. The method of claim 9 further comprising the step of executing the second thread after the swapping of instructions in step (C) by executing at least one instruction in the first instruction buffer.

14. The method of claim 9 further comprising the steps of:

(D) providing a third instruction buffer corresponding to a third thread;

(E) providing a fourth instruction buffer corresponding to a fourth thread; and

(F) swapping instructions stored in the third instruction buffer with instructions stored in the fourth instruction buffer.

15. The method of claim 14 wherein step (F) is performed when the third thread stalls.

16. The method of claim 14 wherein step (F) is performed when the fourth thread stalls.

17. The method of claim 14 wherein the first and third threads simultaneously issue instructions for execution.

18. A method for switching between first and second threads of execution in a multithreaded processor, the method comprising the steps of:

(A) providing a first primary instruction buffer corresponding to the first thread;

(B) providing a second primary instruction buffer corresponding to the second thread;

(C) providing a first backup instruction buffer corresponding to a first backup thread;

(D) providing a second backup instruction buffer corresponding to a second backup thread;

(E) simultaneously issuing instructions from the first primary instruction buffer and from the second primary instruction buffer; and

(F) detecting when one of the first and second primary threads stalls, and in response thereto, swapping instructions stored in one of the first and second primary instruction buffers corresponding to the stalled thread with instructions stored in one of the first and second backup instruction buffers.

19. The method of claim 18 wherein step (E) comprises the steps of:

(1) detecting when the first primary thread stalls, and in response thereto, swapping information stored in the first primary instruction buffer with information stored in the first backup instruction buffer; and

(2) detecting when the second thread stalls, and in response thereto, swapping instructions stored in the second primary instruction buffer with instructions stored in the second backup instruction buffer.

20. The method of claim 18 wherein the first and second backup instruction buffers are part of a pool of backup instruction buffers, wherein instructions in any backup instruction buffer in the pool may be swapped with instructions in the first primary instruction buffer, and wherein instructions in any backup instruction buffer in the pool may be swapped with instructions in the second primary instruction buffer.

21. A method for switching between threads of execution in a multithreaded processor, the method comprising the steps of:

(A) providing a first primary instruction buffer corresponding to the first thread;

(B) providing a second primary instruction buffer corresponding to the second thread;

(C) providing a first backup instruction buffer corresponding to a first backup thread;

(D) providing a second backup instruction buffer corresponding to a second backup thread;

(E) simultaneously issuing instructions from the first primary instruction buffer and from the second primary instruction buffer; and

(F) detecting when the first threads stalls, and in response thereto, swapping instructions stored in the first primary instruction buffer with instructions stored in the first backup instruction buffer, and issuing instructions from the first primary instruction buffer.

22. The method of claim 21 further comprising the step of

(G) detecting when the second thread stalls, and in response thereto, swapping instructions stored in the second primary instruction buffer with instructions stored in the second backup instruction buffer, and issuing from the second primary instruction buffer.

EVIDENCE APPENDIX

An Evidence Appendix is not required for this Appeal Brief.

RELATED PROCEEDINGS APPENDIX

A Related Proceedings Appendix is not required for this Appeal Brief.